

What is claimed is:

1 1. A method in a computer system for flexibly altering software component

2 behavior, the method comprising:

3 intercepting a service request made by a software component;

4 determining a desired behavior for the software component, the desired

5 behavior being specified for the software component and

6 conditions by at least one condition dependent rule; and

7 controlling the software component such that the software component

8 executes the desired behavior.

1 2. The method of claim 1 wherein:

2 intercepting a service request comprises intercepting a software supported

3 system call.

1 3. The method of claim 2 wherein:

2 intercepting a software supported system call further comprises redirecting

3 an entry in an interrupt vector table to alternative code.

1 4. The method of claim 1 wherein:

2 intercepting a service request comprises intercepting a hardware supported

3 system call.

1 5. The method of claim 4 wherein:

intercepting a hardware supported system call further comprises
redirecting an entry in an interrupt vector table to alternative code.

6. The method of claim 1 wherein:

intercepting a service request comprises intercepting a software library
based subroutine call.

7. The method of claim 6 wherein:

intercepting a software library based subroutine call further comprises
modifying at least one dynamically linked library.

8. The method of claim 1 wherein:

intercepting a service request comprises intercepting a subroutine based
service.

9. The method of claim 8 wherein:

intercepting a subroutine based service further comprises redirecting the
subroutine call instruction to alternative code.

10. The method of claim 8 wherein:

intercepting a subroutine based service further comprises patching
machine language entry code of the subroutine.

11. The method of claim 1 wherein:

intercepting a service request comprises intercepting a service dispatch
mechanism based on dynamic name resolution.

1 12. The method of claim 11 wherein:

2 intercepting a service dispatch mechanism based on dynamic name

3 resolution further comprises modifying service lookup name space.

1 13. The method of claim 1 further wherein the desired behavior for the software

2 component further comprises:

3 executing alternative code in response to intercepting the service request.

1 14. The method of claim 13 further comprising:

2 executing alternative code in addition to calling the service request.

1 15. The method of claim 13 wherein:

2 the alternative code performs an operation with a same purpose as that of

3 the service request.

1 16. The method of claim 13 wherein:

2 the alternative code performs an operation with a different purpose from

3 that of the service request.

1 17. The method of claim 1 further wherein the desired behavior for the software

2 component further comprises:

3 preventing execution of the service request.

1 18. The method of claim 17 further comprising:

2 returning a value to the software component so as to simulate execution of
3 the service request, without actually calling the service request.

1 19. The method of claim 1 further wherein the desired behavior for the software
2 component further comprises:

3 preventing code that executes in response to interception of the service
4 request from accessing at least some data.

1 20. The method of claim 19 further comprising:

2 allowing code that executes in response to interception of the service
3 request to access alternative data, different from requested data.

1 21. The method of claim 20 wherein:

2 the alternative data comprises a copy of at least some data.

1 22. The method of claim 19 wherein:

2 code that executes in response to the interception of the service request
3 comprises at least alternative code.

1 23. The method of claim 19 wherein:

2 code that executes in response to the interception of the service request
3 comprises at least the service request.

1 24. The method of claim 1 wherein controlling the software component further
2 comprises:

3 passing alternative parameters to code that executes in response to
4 interception of the service request.

1 25. The method of claim 24 further comprising:
2 creating the alternative parameters by modifying original parameters
3 passed to the service request.

1 26. The method of claim 1 wherein controlling the software component further
2 comprises:
3 executing alternative code in response to interception of the service
4 request.

1 27. The method of claim 1 wherein controlling the software component further
2 comprises:
3 executing alternative code in response to interception of the service
4 request; and
5 executing the service request.

1 28. The method of claim 1 wherein controlling the software component further
2 comprises:
3 preventing execution of the service request.

1 29. The method of claim 1 wherein controlling the software component further
2 comprises:

3 simulating execution of the service request by returning a value to the
4 software component.

1 30. The method of claim 1 wherein controlling the software component further
2 comprises:

3 preventing code that executes in response to interception of the service
4 request from accessing at least some data.

1 31. The method of claim 30 further comprising:

2 allowing code that executes in response to interception of the service
3 request to access alternative data, different from requested data.

1 32. The method of claim 31 wherein:

2 the alternative data comprises a copy of at least some data.

1 33. The method of claim 1 wherein controlling the software component further
2 comprises:

3 returning an alternative value to the software component.

1 34. The method of claim 33 further comprising:

2 creating the alternative value by modifying a value returned by the service
3 request.

1 35. The method of claim 1 further wherein the desired behavior for the software
2 component further comprises:

3 preventing code that executes in response to interception of the service
4 request from accessing a system resource.

1 36. The method of claim 35 wherein:
2 the system resource comprises a network.

1 37. The method of claim 35 wherein:
2 the system resource comprises storage media.

1 38. The method of claim 35 wherein:
2 the system resource comprises a file system.

1 39. The method of claim 35 wherein:
2 the system resource comprises a specific file.

1 40. The method of claim 35 wherein:
2 the system resource comprises configuration information.

1 41. The method of claim 40 wherein:
2 the configuration information comprises registry data.

1 42 The method of claim 1 wherein at least one condition dependant rule that
2 specifies the desired behavior for the software component is static.

1 43 The method of claim 1 wherein dependant rules that specify the desired
2 behavior for the software component comprise a combination of static rules and dynamic
3 rules.

1 44 The method of claim 43 further comprising:
2 modifying the dynamic rules in response to behavior of the software
3 component.

1 45. The method of claim 44 wherein modifying the dynamic rules in response to
2 the behavior of the software component further comprises:
3 responsive to an attempt by the software component to access specific
4 data, creating a rule that specifies that the software component
5 cannot access other data.

1 46. The method of claim 44 wherein modifying the dynamic rules in response to
2 the behavior of the software component further comprises:
3 responsive to an attempt by the software component to access specific
4 data, creating a rule that specifies that the software component
5 cannot perform certain functionality.

1 47. The method of claim 1 wherein the rules that specify the desired behavior for
2 the software component are based on at least one of the following criteria:
3 a user with which the software component is associated;
4 identity of the software component;

5 a time at which the software component is executing;
6 history of the software component;
7 a source of the software component;
8 data which the software component attempts to access;
9 functionality that software component attempts to execute; and
10 computer network resources that the software component attempts to
11 access.

1 48. The method of claim 1 further comprising:

2 specifying at least one rule to control the behavior of at least one software
3 component under at least one condition.

1 49. A computer system for flexibly altering software component behavior, the
2 system comprising:

3 an interception module, for intercepting a service request made by a
4 software component;

5 an altered states engine, for determining a desired behavior for the
6 software component, the desired behavior being specified for the
7 software component and conditions by at least one condition
8 dependent rule, the altered states engine being coupled to the
9 interception module;

10 a static rules database, for storing static condition dependant rules, the
11 static rules database being coupled to the altered states engine;

12 a dynamic rules database, for storing static condition dependant rules, the
13 dynamic rules database being coupled to the altered states engine;
14 and
15 alternative code for executing in response to an intercepted service request
16 made by the software component, for controlling the software
17 component such that the software component executes the desired
18 behavior, the alternative code being coupled to the altered states
19 engine.

1 50. A computer system for flexibly altering software component behavior, the
2 system comprising:

3 a software portion for intercepting a service request made by a software
4 component;

5 a software portion for determining a desired behavior for the software
6 component, the desired behavior being specified for the software
7 component and conditions by at least one condition dependent
8 rule; and

9 a software portion for controlling the software component such that the
10 software component executes the desired behavior.

1 51. A computer system for flexibly altering software component behavior, the
2 system comprising:

3 means for intercepting a service request made by a software component;

4 means for determining a desired behavior for the software component, the
5 desired behavior being specified for the software component and
6 conditions by at least one condition dependent rule; and
7 means for controlling the software component such that the software
8 component executes the desired behavior.

1 52. A computer program product for flexibly altering software component
2 behavior, the computer program product comprising:

3 program code for intercepting a service request made by a software
4 component;
5 program code for determining a desired behavior for the software
6 component, the desired behavior being specified for the software
7 component and conditions by at least one condition dependent
8 rule;
9 program code for controlling the software component such that the
10 software component executes the desired behavior; and
11 a computer readable medium on which the program codes are stored.

1 53. The computer program product of claim 52 further comprising:

2 program code for executing alternative code in response to intercepting the
3 service request.

1 54. The computer program product of claim 52 further comprising:

2 program code for preventing code that executes in response to interception
3 of the service request from accessing at least some data.

1 55. The computer program product of claim 54 further comprising:
2 program code for allowing code that executes in response to interception
3 of the service request to access alternative data, different from
4 requested data.

1 56. The computer program product of claim 52 further comprising:
2 program code for passing alternative parameters to code that executes in
3 response to interception of the service request.

1 57. The computer program product of claim 56 further comprising:
2 program code for creating the alternative parameters by modifying
3 original parameters passed to the service request.

1 58. The computer program product of claim 52 further comprising:
2 program code for executing alternative code in response to interception of
3 the service request.

1 59. The computer program product of claim 58 further comprising:
2 program code for executing alternative code in response to interception of
3 the service request; and
4 program code for executing the service request.

1 60. The computer program product of claim 52 further comprising:
2 program code for preventing execution of the service request.

1 63. The computer program product of claim 52 further comprising:

2 program code for specifying at least one rule to control the behavior of at

3 least one software component under at least one condition.

1 64. A method in a computer system for flexibly altering software component

2 behavior, the method comprising:

3 receiving, by an altered states engine, a service request made by a software

4 component;

5 determining a desired behavior for the software component, the desired

6 behavior being specified for the software component and

7 conditions by at least one condition dependent rule; and

8 controlling the software component such that the software component

9 executes the desired behavior.

1 65. The method of claim 64 further wherein the desired behavior for the software

2 component further comprises:

3 executing alternative code in response to receiving the service request.

1 66. The method of claim 65 further comprising:

2 executing alternative code in addition to calling the service request.

1 67. The method of claim 65 wherein:

2 the alternative code performs an operation with a same purpose as that of

3 the service request.

1 68. The method of claim 65 wherein:

2 the alternative code performs an operation with a different purpose from
3 that of the service request.

1 69. The method of claim 64 further wherein the desired behavior for the software
2 component further comprises:

3 preventing execution of the service request.

1 70. The method of claim 69 further comprising:

2 returning a value to the software component so as to simulate execution of
3 the service request, without actually calling the service request.

1 71. The method of claim 64 further wherein the desired behavior for the software
2 component further comprises:

3 preventing code that executes in response to receipt of the service request
4 from accessing at least some data.

1 72. The method of claim 71 further comprising:

2 allowing code that executes in response to receipt of the service request to
3 access alternative data, different from requested data.

1 73. The method of claim 72 wherein:

2 the alternative data comprises a copy of at least some data.

1 74. The method of claim 71 wherein:

code that executes in response to the receipt of the service request
comprises at least alternative code.

75. The method of claim 71 wherein:

code that executes in response to the receipt of the service request
comprises at least the service request.

76. The method of claim 64 wherein controlling the software component further
comprises:

passing alternative parameters to code that executes in response to receipt
of the service request.

77. The method of claim 76 further comprising:

creating the alternative parameters by modifying original parameters
passed to the service request.

78 The method of claim 64 wherein controlling the software component further
comprises:

executing alternative code in response to receipt of the service request.

79 The method of claim 64 wherein controlling the software component further
comprises:

executing alternative code in response to receipt of the service request;
and
executing the service request.

1 80. The method of claim 64 wherein controlling the software component further
2 comprises:
3 preventing execution of the service request.

1 81. The method of claim 64 wherein controlling the software component further
2 comprises:
3 simulating execution of the service request by returning a value to the
4 software component.

1 82. The method of claim 64 wherein controlling the software component further
2 comprises:
3 preventing code that executes in response to receipt of the service request
4 from accessing at least some data.

1 83. The method of claim 82 further comprising:
2 allowing code that executes in response to receipt of the service request to
3 access alternative data, different from requested data.

1 84. The method of claim 83 wherein:
2 the alternative data comprises a copy of at least some data.

1 85. The method of claim 64 wherein controlling the software component further
2 comprises:
3 returning an alternative value to the software component.

1 86. The method of claim 85 further comprising:
2 creating the alternative value by modifying a value returned by the service
3 request.

1 87. The method of claim 64 further wherein the desired behavior for the software
2 component further comprises:
3 preventing code that executes in response to receipt of the service request
4 from accessing a system resource.

1 88. The method of claim 87 wherein:
2 the system resource comprises a network.

1 89. The method of claim 87 wherein:
2 the system resource comprises storage media.

1 90. The method of claim 87 wherein:
2 the system resource comprises a file system.

1 91. The method of claim 87 wherein:
2 the system resource comprises a specific file.

1 92. The method of claim 87 wherein:
2 the system resource comprises configuration information.

1 93. The method of claim 92 wherein:

2 the configuration information comprises registry data.

1 94. The method of claim 64 wherein at least one condition dependant rule that
2 specifies the desired behavior for the software component is static.

1 95. The method of claim 64 wherein dependant rules that specify the desired
2 behavior for the software component comprise a combination of static rules and dynamic
3 rules.

1 96. The method of claim 95 further comprising:
2 modifying the dynamic rules in response to behavior of the software
3 component.

1 97. The method of claim 96 wherein modifying the dynamic rules in response to
2 the behavior of the software component further comprises:
3 responsive to an attempt by the software component to access specific
4 data, creating a rule that specifies that the software component
5 cannot access other data.

1 98. The method of claim 96 wherein modifying the dynamic rules in response to
2 the behavior of the software component further comprises:
3 responsive to an attempt by the software component to access specific
4 data, creating a rule that specifies that the software component
5 cannot perform certain functionality.

1 99. The method of claim 64 wherein the rules that specify the desired behavior
2 for the software component are based on at least one of the following criteria:

3 a user with which the software component is associated;
4 identity of the software component;
5 a time at which the software component is executing;
6 history of the software component;
7 a source of the software component;
8 data which the software component attempts to access;
9 functionality that software component attempts to execute; and
10 computer network resources that the software component attempts to
11 access.

1 100. The method of claim 64 further comprising:

2 specifying at least one rule to control the behavior of at least one software
3 component under at least one condition.

1 101. A computer system for flexibly altering software component behavior, the
2 system comprising:

3 an receiving module, for receiving a service request made by a software
4 component;
5 an altered states engine, for determining a desired behavior for the
6 software component, the desired behavior being specified for the
7 software component and conditions by at least one condition

8 dependent rule, the altered states engine being coupled to the
9 receiving module;
10 a static rules database, for storing static condition dependant rules, the
11 static rules database being coupled to the altered states engine;
12 a dynamic rules database, for storing static condition dependant rules, the
13 dynamic rules database being coupled to the altered states engine;
14 and
15 alternative code for executing in response to a received service request
16 made by the software component, for controlling the software
17 component such that the software component executes the desired
18 behavior, the alternative code being coupled to the altered states
19 engine.

1 102. A computer system for flexibly altering software component behavior, the
2 system comprising:

3 a software portion for receiving a service request made by a software
4 component;
5 a software portion for determining a desired behavior for the software
6 component, the desired behavior being specified for the software
7 component and conditions by at least one condition dependent
8 rule; and
9 a software portion for controlling the software component such that the
10 software component executes the desired behavior.

1 103. A computer system for flexibly altering software component behavior, the
2 system comprising:

3 means for receiving a service request made by a software component;

4 means for determining a desired behavior for the software component, the
5 desired behavior being specified for the software component and
6 conditions by at least one condition dependent rule; and

7 means for controlling the software component such that the software
8 component executes the desired behavior.

1 104. A computer program product for flexibly altering software component
2 behavior, the computer program product comprising:

3 program code for receiving a service request made by a software
4 component;

5 program code for determining a desired behavior for the software
6 component, the desired behavior being specified for the software
7 component and conditions by at least one condition dependent
8 rule;

9 program code for controlling the software component such that the
10 software component executes the desired behavior; and
11 a computer readable medium on which the program codes are stored.

1 105. The computer program product of claim 104 further comprising:

2 program code for executing alternative code in response to receiving the
3 service request.

1 106. The computer program product of claim 104 further comprising:

2 program code for preventing code that executes in response to receipt of
3 the service request from accessing at least some data.

1 107. The computer program product of claim 106 further comprising:

2 program code for allowing code that executes in response to receipt of the
3 service request to access alternative data, different from requested
4 data.

1 108. The computer program product of claim 104 further comprising:

2 program code for passing alternative parameters to code that executes in
3 response to receipt of the service request.

1 109. The computer program product of claim 108 further comprising:

2 program code for creating the alternative parameters by modifying
3 original parameters passed to the service request.

1 110. The computer program product of claim 104 further comprising:

2 program code for executing alternative code in response to receipt of the
3 service request.

1 111. The computer program product of claim 110 further comprising:

2 program code for executing alternative code in response to receipt of the
3 service request; and
4 program code for executing the service request.

1 112. The computer program product of claim 104 further comprising:
2 program code for preventing execution of the service request.

1 113. The computer program product of claim 104 further comprising:
2 program code for simulating execution of the service request by returning
3 a value to the software component.

1 114. The computer program product of claim 104 further comprising:
2 program code for returning an alternative value to the software
3 component.

1 115. The computer program product of claim 104 further comprising:
2 program code for specifying at least one rule to control the behavior of at
3 least one software component under at least one condition.